



The interpolated variational iteration method for solving a class of nonlinear optimal control problems

Mohammad Shirazian*

Abstract

Despite the variety of methods available to solve nonlinear optimal control problems, numerical methods are still evolving to solve these problems. This paper deals with the numerical solution of nonlinear optimal control affine problems by the interpolated variational iteration method, which was introduced in 2016 to improve the variational iteration method. For this purpose, the optimality conditions are first derived as a two-point boundary value problem and then converted to an initial value problem with the unknown initial values for costates. The speed and convergence of the method are compared with the existing methods in the form of three examples, and the initial values of the costates are obtained by an efficient technique in each iteration.

AMS subject classifications (2020): Primary 45D05; Secondary 42C10, 65G99.

Keywords: Optimal control problems; Interpolated variational iteration method; Pontryagin's maximum principle.

1 Introduction

One of the important problems in the field of engineering [13, 6], robotics [14, 11], biology [1, 16], and so on, is finding the best control strategy to use available resources optimally. In practice, most of these problems are formulated nonlinearly, which in general do not have analytical solutions, and

*Corresponding author

Received 27 June 2022; revised 15 September 2022; accepted 8 October 2022

Mohammad Shirazian

Department of Mathematics, University of Neyshabur, Neyshabur, Iran. e-mail: shirazian@neyshabur.ac.ir

one must look for reliable numerical solutions for them. Generally, consider the following nonlinear control system:

$$\begin{aligned} \dot{x}(t) &= A(x(t)) + B(t)u(t), & t \in [t_0, t_f], \\ x(t_0) &= x_0, \quad x(t_f) = x_f, \end{aligned} \quad (1)$$

where $x(t)$ and $u(t)$ are n - and m -vector of state and control functions, respectively. The aim is to minimize the objective functional

$$J[x, u] = \frac{1}{2} \int_{t_0}^{t_f} (x^T(t)Q(t)x(t) + u^T(t)R(t)u(t))dt, \quad (2)$$

subject to the dynamical system (1). Moreover, $Q(t)$ is a positive semi-definite real matrix, and $R(t)$ is a positive definite real matrix.

There are two approaches to solving this problem. One is the direct method of parameterizing the state and control functions and solving a nonlinear optimization problem. The other is the indirect method of using optimality conditions, derived by the Pontryagin maximum principle (PMP) or Hamilton–Jacobi–Bellman equation [3, 15]. The main topic of our discussion in this paper is the optimality conditions derived by PMP, which results in the following two-point boundary value problem (TPBVP) as

$$\begin{aligned} \dot{x}(t) &= A(x(t)) + B(t)u(t), \\ \dot{\lambda}(t) &= -Q(t)x(t) - A_x^T(x(t))\lambda(t), \\ x(t_0) &= x_0, x(t_f) = x_f, \end{aligned} \quad (3)$$

where $\lambda(t)$ is an n -vector of co-state functions, $A_x^T(x(t)) = \nabla_x A^T(x(t))$, and the optimal control is given by

$$u(t) = -R^{-1}B^T(t)\lambda(t). \quad (4)$$

In 2012, Berkani, Manseur, and Maldi [4] used the variational iteration method (VIM) to solve this problem, for which the control variable can be calculated in terms of state variables. They then turned the problem into a variational problem and solved it with VIM. It is noteworthy that the VIM is an analytical-approximate method for solving initial value problems (IVPs), which was introduced by He [8] in 1999, and many researchers developed the method to increase its efficiency and validity in recent years [20, 5]. For solving (3), the author and Effati [24] used a combination of the shooting method with VIM. However, the time-consuming and complex integration of nonlinear functions reduces its efficiency, at each iteration.

Matinfar and Saeidy [17] applied a modified VIM (MVIM) that uses He polynomials. Simultaneously, Saberi Nik, Effati, and Yildirim [22] used a differential transformation method along with differential transformation polynomials to solve (3). A comparison between the existing methods,

MVIM, Homotopy perturbation method (HPM), and Adomian decomposition method was done in 2016 by Jafari. Ghasempour, and Baleanu [9].

Mirhosseini-Alizamini and Effati [19] extended the VIM for optimal delay control problems. In 2020, the author and Effati [25] used the successive approximation method (SAM) to solve (3), which has a very good computational speed but does not guarantee the convergence for problems with large domains.

An improvement of VIM, called Interpolated VIM (IVIM) introduced by Salkuyeh and Tavakoli [23], interpolates the VIM with the well-known spline functions, which make the method suitable for approximating all types of $L^2[t_0, t_f]$ solutions, especially for control and state functions. This property motivates us to apply the IVIM for solving optimal control problems. This method has low computational time in comparison with VIM since it uses an approximation for integration. It also maintains accuracy and convergence.

This paper is organized as follows. In Section 2, the basic ideas of IVIM are briefly described. Then, the application of IVIM for solving nonlinear optimal control problems (NOCPs) is explained in Section 3. Several numerical examples are illustrated in Section 4, which show the accuracy and efficiency of the proposed method compared to the existing ones.

2 Basic ideas of the interpolated variational iteration method

As stated earlier, the VIM is an analytical-approximate iterative method for solving IVPs [8]. To explain this method, consider the following one-dimensional IVP

$$\begin{aligned} \dot{y}(t) &= f(t, y(t)), \quad t \in [0, T], \\ y(0) &= y^0, \end{aligned} \quad (5)$$

where $y(\cdot) \in \mathbb{R}$ and f satisfies the Lipschitz condition:

$$|f(t, y(t)) - f(t, \tilde{y}(t))| \leq L|y(t) - \tilde{y}(t)|, \quad (6)$$

for any $t \in [0, T]$ and for some Lipschitz constant $L > 0$. We make the following iterative formula, which is known as the VIM main formula:

$$y_{m+1}(t) = y_m(t) + \int_0^t \Lambda(s, t) (\dot{y}_m(s) - f(s, y_m(s))) ds, \quad m = 0, 1, 2, \dots, \quad (7)$$

where $\Lambda(s, t)$ is known as a Lagrange multiplier [2, 28], and $y_0(t)$ should be chosen such that it satisfies $y_0(0) = y^0$. The VIM formula produces a sequence of functions $\{y_m(t)\}$, which converges to the exact solution of (5), under suitable conditions and does not require any linearization or perturbation. Due to the attractiveness of the method for calculating accurate solutions, many researchers have used it to solve difficult problems such as

dispersive optimal solitons [7], coupled Burgers' equation [26], fractional option pricing [18], fractal Fisher's equation [27]. However, the integration of nonlinear function $f(t, y(t))$ in all these researches may greatly reduce the speed of the method or lead to the failure of the method. To overcome this problem, first, we apply the integration by parts for (7) to get:

$$y_{m+1}(t) = G_m(t) - \int_0^t H_m(s, t) ds, \quad (8)$$

where

$$G_m(t) = (1 + \Lambda(t, t))y_m(t) - \Lambda(0, t)y_m(0),$$

and

$$H_m(s, t) = \frac{\partial \Lambda}{\partial s}(s, t)y_m(s) + \Lambda(s, t)f(s, y_m(s)).$$

Now, we interpolate $y_m(t)$ by first order B-spline functions [12], at equidistance points t_i , $i = 1, \dots, n$ on $[0, T]$. For any $i = 2, 3, \dots, n - 1$, let us define

$$\mathcal{B}_i(t) = \frac{t - t_{i-1}}{h} \chi_{[t_{i-1}, t_i)}(t) + \frac{t_{i+1} - t}{h} \chi_{[t_i, t_{i+1})}(t),$$

and

$$\mathcal{B}_n(t) = \frac{t - t_{n-1}}{h} \chi_{[t_{n-1}, t_n)}(t),$$

where $\chi_{\mathcal{A}}$ is the characteristic function defined on a set \mathcal{A} . Its value is 1 on \mathcal{A} and is 0 otherwise. It is obvious from the definition that $\mathcal{B}_i(t_i) = 1$ and $\mathcal{B}_i(t_j) = 0$, for $j \neq i$. Then, we employ the interpolant

$$\hat{y}_m(t) = \sum_{i=2}^n y_m(t_i) \mathcal{B}_i(t).$$

Thus, $\hat{y}_m(t_i) = y_m(t_i)$, for $i = 2, \dots, n$. Now, we substitute the interpolant \hat{y}_m into (7) and after some simplifications, we arrive at

$$\begin{aligned} y_{m+1}(t_i) &= \hat{y}_{m+1}(t_i) \\ &= G_m(t_i) - h \sum_{r=2}^{i-1} H_m(t_r, t_i) - \frac{h}{2} (H_m(t_1, t_i) + H_m(t_i, t_i)). \end{aligned} \quad (9)$$

This is the main formula of IVIM, which overcomes the nonlinear integration of VIM and will be used as a tool for solving NOCPs in the next section.

Now, we state the convergence of VIM and IVIM sequences as the next two theorems, which were proved in [23].

Theorem 1. Let f be a continuous function on $[0, T]$. If $y_m(t) \in \mathcal{C}^1([0, T])$ for $m = 0, 1, 2, \dots$, then the VIM sequence (7) converges to the exact solution of (5).

Theorem 2. Let f be a two times continuously differentiable function. If the VIM sequence (7) is to the exact solution of (5), then so is the IVIM sequence (9).

3 The application of IVIM for solving NOCPs

For the sake of simplicity, we rewrite the TPBVP in (3) as

$$\begin{aligned} \dot{x} &= \mathcal{F}(t, x, \lambda), \\ \dot{\lambda} &= \mathcal{G}(t, x, \lambda), \\ x(t_0) &= x_0, \quad x(t_f) = x_f, \end{aligned} \quad (10)$$

where

$$\begin{aligned} \mathcal{F}(t, x, \lambda) &= A(x(t)) - B(t)R^{-1}B^T(t)\lambda(t), \\ \mathcal{G}(t, x, \lambda) &= -Q(t)x(t) - A_x^T(x(t))\lambda(t). \end{aligned}$$

This nonlinear TPBVP has no analytical solution in general. Thus, researchers are looking for reliable and almost accurate solutions to it. The technique we are studying here is IVIM, which was described in Section 2. Given that the above problem is TPBVP, we first convert it to an IVP as follows [24]:

$$\begin{aligned} \dot{x} &= \mathcal{F}(t, x, \lambda), \\ \dot{\lambda} &= \mathcal{G}(t, x, \lambda), \quad t \in [t_0, t_f] \\ x(t_0) &= x_0, \quad \lambda(t_0) = \alpha, \end{aligned} \quad (11)$$

where α is an n -vector unknown variable. This variable could be identified after sufficient iterations of IVIM.

As Section 2, we create the VIM correction functional as follows:

$$x_{m+1}(t) = x_m(t) + \int_{t_0}^t \Lambda_1(t, s) \{ \dot{x}_m(s) - \mathcal{F}(s, x_m(s), \lambda_m(s)) \} ds, \quad (12)$$

$$\lambda_{m+1}(t) = \lambda_m(t) + \int_{t_0}^t \Lambda_2(t, s) \{ \dot{\lambda}_m(s) - \mathcal{G}(s, x_m(s), \lambda_m(s)) \} ds, \quad (13)$$

with $x_0(t) = x_0$, $\lambda_0(t) = \alpha$ and $m \geq 0$. To determine general multipliers $\Lambda_1(t, s)$ and $\Lambda_2(t, s)$, we restrict the variations for \mathcal{F} and \mathcal{G} , that is, $\delta\tilde{\mathcal{F}} = \delta\tilde{\mathcal{G}} = 0$. Therefore, the optimal multipliers become $\Lambda_1(t, s) = \Lambda_2(t, s) = -1$. Compared with (8), consider $G_1(t)$ and $H_1(t)$ corresponding to $x(t)$ and

$$\begin{aligned} G_{1,m}(t) &= (1 + \Lambda_1(t, t))x_m(t) - \Lambda_1(0, t)x_0, \\ H_{1,m}(s, t) &= \frac{\partial \Lambda_1}{\partial s}(s, t)x_m(s) + \Lambda_1(s, t)\mathcal{F}(s, x_m(s), \lambda_m(s)). \end{aligned}$$

Similarly, consider $G_2(t)$ and $H_2(t)$ corresponding to $\lambda(t)$ and

$$\begin{aligned} G_{2,m}(t) &= (1 + \Lambda_2(t, t))x_m(t) - \Lambda_2(0, t)\alpha, \\ H_{2,m}(s, t) &= \frac{\partial \Lambda_2}{\partial s}(s, t)x_m(s) + \Lambda_2(s, t)\mathcal{G}(s, x_m(s), \lambda_m(s)). \end{aligned}$$

For a given $n \in \mathbb{N}$, assume $t_i = t_0 + (i - 1)h$, $i = 2, 3, \dots$, where $h = \frac{t_f - t_0}{n-1}$. Then, the B-spline interpolants of $x_{m+1}(t)$ and $\lambda_{m+1}(t)$ become

$$\begin{aligned} x_{m+1}^{(h)}(t) &= \sum_{i=2}^n \hat{x}_{m+1}(t_i)\mathcal{B}_i(t), \\ \lambda_{m+1}^{(h)}(t) &= \sum_{i=2}^n \hat{\lambda}_{m+1}(t_i)\mathcal{B}_i(t), \end{aligned}$$

where the coefficients are calculated as

$$\begin{aligned} x_{m+1}(t_i) &= \hat{x}_{m+1}(t_i) \\ &= G_{1,m}(t_i) - h \sum_{r=2}^{i-1} H_{1,m}(t_r, t_i) - \frac{h}{2} (H_{1,m}(t_1, t_i) + H_{1,m}(t_i, t_i)), \end{aligned} \tag{14}$$

$$\begin{aligned} \lambda_{m+1}(t_i) &= \hat{\lambda}_{m+1}(t_i) \\ &= G_{2,m}(t_i) - h \sum_{r=2}^{i-1} H_{2,m}(t_r, t_i) - \frac{h}{2} (H_{2,m}(t_1, t_i) + H_{2,m}(t_i, t_i)), \end{aligned} \tag{15}$$

for any $i = 2, 3, \dots, n$. We call these two equations as the IVIM main formulas. The convergence of the IVIM is a straightforward conclusion of Theorems 1–2, which we state in the next theorem.

Theorem 3. Let \mathcal{F} and \mathcal{G} be two times continuously differentiable functions with respect to their first variable and let they satisfy the Lipschitz condition. If $x_m(t), \lambda_m(t) \in \mathbb{C}^1([t_0, t_f])$, then the IVIM of (14)–(15) is convergent to the exact solutions of (11), say $x^*(t, \alpha)$ and $\lambda^*(t, \alpha)$. Accordingly, $x^*(t, \alpha^*)$, $\lambda^*(t, \alpha^*)$ are the exact solution of (10) when α^* is the real root of $x^*(t_f, \alpha) = x_f$.

Proof. Since \mathcal{F} and \mathcal{G} are continuous on $[t_0, t_f]$, and $x_m(t), \lambda_m(t) \in \mathbb{C}^1([t_0, t_f])$, Theorem 3 guarantees the convergence of the VIM formulas (12)–(13) to the exact solution of (11). Besides, since \mathcal{F} and \mathcal{G} are two times continuously differentiable functions, Theorem 4 ensures that the IVIM is convergent to the exact solution of (11), say $x^*(t, \alpha)$ and $\lambda^*(t, \alpha)$. Clearly, for every α , $x^*(t_0, \alpha) = x_0$ and $\lambda^*(t_0, \alpha) = \alpha$. Then, from (10), we have $x(t_f) = x_f = x^*(t_f, \alpha)$. Therefore, to get the exact solution of (10), one should solve $x^*(t_f, \alpha) = x_f$ for some appropriate real α . \square

Theorem 4. Under the assumptions of Theorem 3, the sequences $\{u_m(t)\}$ and $\{J_m\}$ defined by

$$u_m(t) = -R^{-1}B^T(t)\lambda_m(t), \quad (16)$$

$$J_m = \frac{h}{2}(\mathcal{L}(x_0, u_m(t_1)) + \mathcal{L}(x_m(t_f), 0)) + h \sum_{i=2}^{n-1} \mathcal{L}(x_m(t_i), u_m(t_i)), \quad (17)$$

converge to the optimal control and optimal objective value, respectively, where $\mathcal{L}(x, u) = \frac{1}{2}(x^T Q x + u^T R u)$.

Proof. Taking limit from (16) and putting $\alpha = \alpha^*$ yield

$$\begin{aligned} u^*(t) &:= \lim_{m \rightarrow \infty} u_m(t) = -R^{-1}B^T(t) \lim_{m \rightarrow \infty} \lambda_m(t) \\ &= -R^{-1}B^T(t)\lambda^*(t, \alpha^*), \end{aligned}$$

where $u^*(t)$ is the optimal control function, since $x^*(t, \alpha^*)$ and $\lambda^*(t, \alpha^*)$ are the exact solutions of extreme conditions in (10).

Now, taking limit from (17), as $n, m \rightarrow \infty$, and using the trapezoidal integration rule result in

$$\begin{aligned} &\lim_{m \rightarrow \infty} \lim_{n \rightarrow \infty} J_{m,n} \\ &= \lim_{m \rightarrow \infty} \lim_{n \rightarrow \infty} \frac{h}{2}(\mathcal{L}(x_0, u_m(t_1)) + \mathcal{L}(x_m(t_f), 0)) + h \sum_{i=2}^{n-1} \mathcal{L}(x_m(t_i), u_m(t_i)) \\ &= \lim_{m \rightarrow \infty} \int_{t_0}^{t_f} \mathcal{L}(x_m(t), u_m(t)) dt \\ &= \lim_{m \rightarrow \infty} \frac{1}{2} \int_{t_0}^{t_f} (x_m(t)Q(t)x_m(t) + u_m(t)^T R(t)u_m(t)) dt \\ &= \frac{1}{2} \int_{t_0}^{t_f} \left(\lim_{m \rightarrow \infty} x_m(t)Q(t) \lim_{m \rightarrow \infty} x_m(t) + \lim_{m \rightarrow \infty} u_m^T(t)R(t) \lim_{m \rightarrow \infty} u_m(t) \right) dt \\ &= \frac{1}{2} \int_{t_0}^{t_f} (x^*(t, \alpha^*)Q(t)x^*(t, \alpha^*) + u^*(t)^T R u^*(t)) dt \\ &= J^*. \end{aligned}$$

It follows that J^* is the optimal objective value and the proof is completed. \square

Now, we present an algorithm to summarize our proposed method. We choose the stopping criterion

$$\bar{E}_m = \left| \frac{J_m - J_{m-1}}{J_m} \right| < \epsilon, \quad (18)$$

which maintains the accuracy of the objective value for any sufficiently small tolerance limit $\epsilon > 0$.

Algorithm of IVIM for solving nonlinear OCPs

Step 1. Let $m := 1$, $x_0(t) := x_0$, $\lambda_0(t) := \alpha$ and $\epsilon > 0$.

Step 2. Calculate state and costate functions, $x_{m+1}(t, \alpha)$ and $\lambda_{m+1}(t, \alpha)$, through the IVIM formulas (14)–(15).

Step 3. Solve $x_{m+1}(t_f, \alpha) = x_f$ for α , to get α^* .

Step 4. Find the suboptimal control $u_{m+1}(t, \alpha^*)$ and the suboptimal objective value J_{m+1} , by (16)–(17).

Step 5. If $\bar{E}_{m+1} < \epsilon$, then goto **Step 6**; otherwise let $m := m + 1$ and goto **Step 2**.

Step 6. Stop the Algorithm. $u_{m+1}(t)$ and J_{m+1} are the desired optimal control and objective value.

Remark 1. If the state function has free final condition, then the costate final condition $\lambda(t_f) = 0$ must be applied. Therefore, to calculate α in **Step 3**, the equation $\lambda_{m+1}(t_f, \alpha) = 0$ must be used instead of $x_{m+1}(t_f, \alpha) = x_f$.

4 Illustrative examples

In this section, we give two nonlinear examples to illustrate the reliability and efficiency of our proposed method. In the first example, we describe the algorithm process in detail. Also, we present the second the third example to compare the method with some other existing methods. Since there are no exact solutions for these three examples, we use criterion (18) to stop the algorithm and check the accuracy of the method. Besides, we use the Maple “dsolve” package as a reliable solution for comparison. The codes were developed using symbolic software Maple 18 on a machine with Intel Core 2 Due Processor 2.53 GHz and 4 GB RAM.

Example 1. Consider the following nonlinear control system [24, 25]:

$$\begin{aligned} \dot{x} &= \frac{1}{2}x^2(t) \sin x(t) + u(t), & t \in [0, 1], \\ x(0) &= 0, \quad x(1) = 0.5, \end{aligned}$$

under which the following objective functional should be minimized:

$$J = \int_0^1 u^2(t) dt.$$

Conforming to (3), we have

$$\dot{x}(t) = \frac{1}{2}x^2(t) \sin x(t) - \frac{1}{2}\lambda(t), \quad (19)$$

$$\dot{\lambda}(t) = -\lambda(t)x(t) \sin x(t) - \frac{1}{2}\lambda(t)x^2(t) \cos x(t), \quad t \in [0, 1] \quad (20)$$

$$x(0) = 0, \quad x(1) = 0.5, \quad (21)$$

and the control function is $u(t) = -\frac{1}{2}\lambda(t)$. We substitute the final state condition $x(1) = 0.5$, by $\lambda(0) = \alpha$ to get

$$\begin{aligned} \dot{x}(t) &= \frac{1}{2}x^2(t) \sin x(t) - \frac{1}{2}\lambda(t), \\ \dot{\lambda}(t) &= -\lambda(t)x(t) \sin x(t) - \frac{1}{2}\lambda(t)x^2(t) \cos x(t), \quad t \in [0, 1], \\ x(0) &= 0, \quad \lambda(0) = \alpha, \end{aligned}$$

where α is an unknown real variable. In view of (11), \mathcal{F} and \mathcal{G} are the right-hand sides of (19)–(20), respectively. They are analytic functions with respect to x and λ , and they satisfy the Lipschitz condition (6), with $L_{\mathcal{F}} = 9/16$ and $L_{\mathcal{G}} = 5/16$, whenever $|x| \leq 0.5$ and $\lambda \leq 1$. Hence, Theorem 3 guarantees the convergence of both VIM and IVIM formulas. The VIM formulas (12)–(13) become

$$\begin{aligned} x_{m+1}(t) &= x_m(t) - \int_0^t \left\{ \dot{x}_m(s) - \frac{1}{2}x_m^2(s) \sin x_m(s) + \frac{1}{2}\lambda_m(s) \right\} ds, \\ \lambda_{m+1}(t) &= \lambda_m(t) - \int_0^t \left\{ \dot{\lambda}_m(s) + \lambda_m(s)x_m(s) \sin x_m(s) \right. \\ &\quad \left. + \frac{1}{2}\lambda_m(s)x_m^2(s) \cos x_m(s) \right\} ds, \\ x_0(t) &= 0, \quad \lambda_0(t) = \alpha, \quad m \geq 0. \end{aligned}$$

Now, interpolating VIM at $t_j = 0, \frac{1}{2}, 1$, the IVIM formulas (14)–(15) give the following results, after the first iteration:

$$\begin{aligned} x_1(t_j) &= 0, \quad -\frac{1}{4}\alpha - \frac{1}{128}\alpha^2 \sin\left(\frac{1}{4}\alpha\right), \quad -\frac{1}{64}\alpha^2 \sin\left(\frac{1}{4}\alpha\right) - \frac{1}{2}\alpha - \frac{1}{32}\alpha^2 \sin\left(\frac{1}{2}\alpha\right), \\ \lambda_1(t_j) &= \alpha, \quad \alpha - \frac{1}{16}\alpha^2 \sin\left(\frac{1}{4}\alpha\right) - \frac{1}{128}\alpha^3 \cos\left(\frac{1}{4}\alpha\right), \\ \alpha - \frac{1}{8}\alpha^2 \sin\left(\frac{1}{4}\alpha\right) - \frac{1}{64}\alpha^3 \cos\left(\frac{1}{4}\alpha\right) - \frac{1}{8}\alpha^2 \sin\left(\frac{1}{2}\alpha\right) - \frac{1}{32}\alpha^3 \cos\left(\frac{1}{2}\alpha\right). \end{aligned}$$

The final state condition, $x_1(1) = 0.5$, results in

$$-\frac{1}{64}\alpha^2 \sin\left(\frac{1}{4}\alpha\right) - \frac{1}{2}\alpha - \frac{1}{32}\alpha^2 \sin\left(\frac{1}{2}\alpha\right) = 0.5, \quad (22)$$

Table 1: Calculation of α for $n = 3$ and $m = 1$, Example 1.

α	Objective functional
-0.9659445089	0.2146370242
-1.023796574	0.2387946125

which obtains two real roots. These real roots along with their corresponding objective values are summarized in Table 1. We suggest choosing the parameter α , which gives a better objective value, that is, $\alpha = -0.9659445089$. This parameter could be a good starting point for root-finding iterative methods, such as Newton, at the next iteration of the IVIM.

Table 2 shows the results of IVIM procedure up to $m = 5$ iterations with $n = 10$ nodes.

Table 2: Simulation results for $n = 10$, Example 1.

m	J_m	\bar{E}_m	CPU time	α
1	0.22264786	-	0.000	-0.9717459824
2	0.23465678	5.1177E-02	0.016	-0.9985940802
3	0.23541609	3.2254E-03	0.078	-0.9997821239
4	0.23513137	1.2109E-03	0.359	-0.9991493164
5	0.23512939	8.4519E-06	2.266	-0.9991493164

To compare the IVIM with two recent methods, VIM [24] and Modal series [10], we summarize their results in Table 3 after five iterations with the same stopping criterion (18), with $\epsilon = 2 \times 10^{-5}$. Clearly, the IVIM converges rapidly, in contrast with VIM and Modal series, since it reaches a better tolerance limit. Also, the suboptimal control and state of the proposed VIM and Modal series methods are demonstrated in Figure 1.

Table 3: Comparing results of IVIM, VIM, Modal series, Example 1.

Method	J_m	\bar{E}_m
IVIM (m=5)	0.2351	8.5E-06
VIM (m=5)	0.2353	1.7E-03
Modal series method (m=5)	0.2346	1.3E-03

Example 2. Consider the nonlinear control system described by

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1 x_2, \\ \dot{x}_2 &= -x_1 + x_2 + x_2^2 + u, \\ x_1(0) &= -0.8, \quad x_2(0) = 0, \end{aligned} \quad (23)$$

and the objective functional

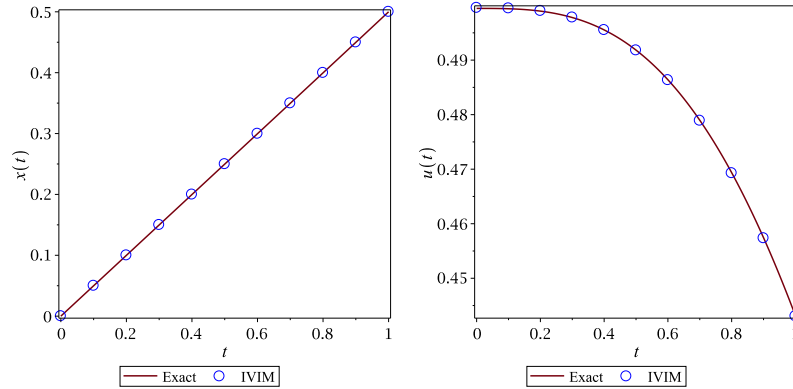


Figure 1: Suboptimal state and control functions, Example 1.

$$J = \frac{1}{2} \int_0^1 (x_1^2 + x_2^2 + u^2) dt. \tag{24}$$

One can obtain the optimality conditions as

$$\begin{aligned} \dot{\lambda}_1 &= -(x_1 + \lambda_1 x_2 - \lambda_2), \\ \dot{\lambda}_2 &= -(x_2 + \lambda_1(1 + x_1) + \lambda_2(1 + 2x_2)), \\ \dot{x}_1 &= x_2 + x_1 x_2, \\ \dot{x}_2 &= -x_1 + x_2 + x_2^2 - \lambda_2, \\ x_1(0) &= -0.8, \quad x_2(0) = 0, \end{aligned}$$

and the optimal control law as $u = -\lambda_2$. Since $x_1(t)$ and $x_2(t)$ are free at $t = 1$, we use the Remark 1 and put $\lambda_1(1) = \lambda_2(1) = 0$.

Now, due to the polynomial nature of the system and their validity in the assumptions of Theorem 3, we apply the proposed algorithm for $\epsilon = 10^{-3}$ to get an accurate optimal objective value. It is worth noting that for some iterations, $m = 2$, solving $\lambda_{1,m}(1, \alpha) = 0$ and $\lambda_{2,m}(1, \alpha) = 0$, results in different values of α . In these cases, as shown in Table 4, we clearly select the vector α that yields the minimum objective functional, that is, $\alpha_1 = -1.227402971$ and $\alpha_2 = 0.5053886518$.

Table 4: Calculation of α for $m = 2$, Example 2.

α_1	α_2	Objective functional
-1.227402971	0.5053886518	0.4113400952
4.538547207	-1.207206466	2.664175385
1.875104138	-1.907110672	4.484594474

Table 5: Simulation results for different iterations, Example 2.

m (Itr.)	VIM			SAM			IVIM		
	J_m	E_m	Time	J_m	E_m	Time	J_m	E_m	Time
1	0.8533	-	0.031	0.5357	-	0.015	0.8560	-	0
2	0.4108	1.1E-0	0.031	0.7575	2.9E-1	0.031	0.4113	1.1E-0	0.015
3	0.4325	5.0E-2	0.046	0.9930	2.4E-1	0.063	0.4340	5.2E-2	0.078
4	0.4395	1.6E-2	0.125	0.6750	4.7E-1	0.046	0.4413	1.7E-2	0.124
5	0.4422	6.1E-3	1.544	0.3503	9.3E-1	0.094	0.4443	6.9E-3	0.296
6	0.4430	1.8E-3	45.693	0.4245	1.7E-1	0.078	0.4453	2.3E-3	1.279
7	-	-	>1800	0.6025	3.0E-1	0.078	0.4456	6.1E-4	7.504

Then, one can use these α_1 and α_2 as initial guesses for solving $\lambda_{1,3}(1, \alpha) = 0$ and $\lambda_{2,3}(1, \alpha) = 0$. This technique greatly improves system speed and reduces computational complexity.

The objective values, relative errors and the CPU time of the algorithm iterations are summarized in Table 5. Moreover, to clarify the reliability of the algorithm, we compare the results with VIM [24] and SAM [25]. As shown in Table 5, SAM has a divergent behavior, and although it has less computational time than the other two methods, it does not reach the desired tolerance limit. Also, the VIM requires more than 1800 seconds of computational time for $m = 7$ and does not reach the desired tolerance. This is while the IVIM reaches the desired tolerance limit after $m = 7$ iterations, and it only takes 7.504 seconds of CPU time.

From the beginning of the algorithm, if we considered $\epsilon = 5 \times 10^{-3}$, both IVIM and VIM would get the desired solutions after $m = 6$ iterations, but IVIM in 1.279 seconds and VIM in 45.693 seconds of CPU time, which clearly shows the speed of convergence of IVIM.

Figure 2 shows the solutions after $m = 7$ algorithm iterations, in comparison with “dsolve” package of Maple as the exact solutions.

Example 3. Consider the optimal maneuvers of a rigid asymmetric spacecraft given by [21]

$$\begin{aligned}
 \dot{x}_1(t) &= -\frac{I_3 - I_2}{I_1} x_2(t)x_3(t) + \frac{1}{I_1} u_1(t), \\
 \dot{x}_2(t) &= -\frac{I_1 - I_3}{I_2} x_1(t)x_3(t) + \frac{1}{I_2} u_2(t), \\
 \dot{x}_3(t) &= -\frac{I_2 - I_1}{I_3} x_1(t)x_2(t) + \frac{1}{I_3} u_3(t),
 \end{aligned} \tag{25}$$

with the boundary conditions

$$\begin{aligned}
 x_1(0) &= 0.01 \text{ r/s}, & x_2(0) &= 0.005 \text{ r/s}, & x_3(0) &= 0.001 \text{ r/s}, \\
 x_1(100) &= x_2(100) = x_3(100) &= 0 \text{ r/s},
 \end{aligned} \tag{26}$$

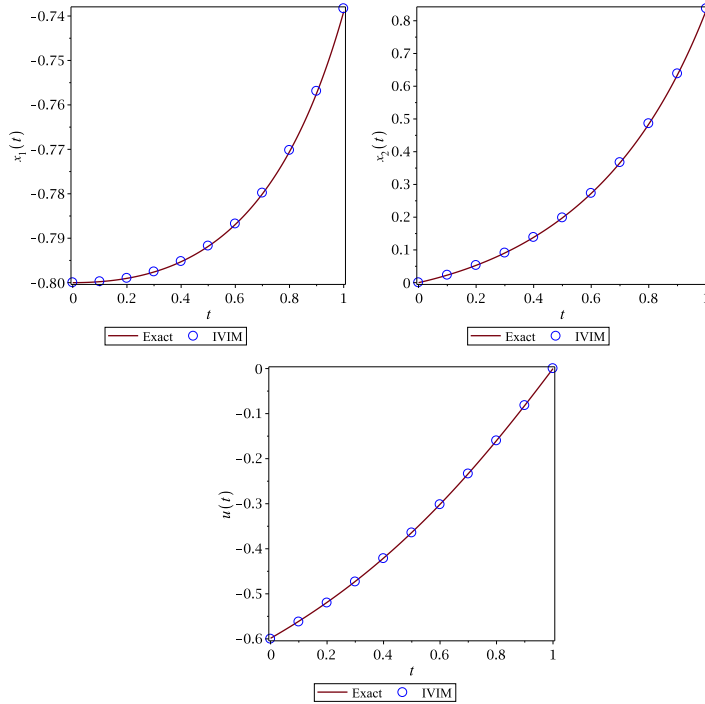


Figure 2: The optimal trajectories and control of Example 2.

where x_1 , x_2 , and x_3 are angular velocities of the spacecraft, u_1 , u_2 , and u_3 are control torques, $I_1 = 86.24$, $I_2 = 85.07$, and $I_3 = 113.59$ kg m² are the spacecraft principle inertia. The controls $u_i(t), i = 1, 2, 3$, should be chosen to minimize the following quadratic performance index:

$$J[x, u] = \frac{1}{2} \int_0^{100} (u_1^2(t) + u_2^2(t) + u_3^2(t))dt, \tag{27}$$

In view of (1)–(2), we have:

$$A(x(t)) = \begin{bmatrix} -\frac{I_3 - I_2}{I_1} x_2(t)x_3(t) \\ -\frac{I_1 - I_3}{I_2} x_1(t)x_3(t) \\ -\frac{I_2 - I_1}{I_3} x_1(t)x_2(t) \end{bmatrix}.$$

Also, $B(t) = \text{diag}(\frac{1}{I_1}, \frac{1}{I_2}, \frac{1}{I_3})$, $Q(t) = O_{3 \times 3}$, and $R(t) = I_{3 \times 3}$. Then, the optimality conditions become

$$\dot{x}_1(t) = -\frac{I_3 - I_2}{I_1} x_2(t)x_3(t) - \frac{\lambda_1(t)}{I_1^2},$$

$$\begin{aligned} \dot{x}_2(t) &= -\frac{I_1 - I_3}{I_2} x_1(t)x_3(t) - \frac{\lambda_2(t)}{I_2^2}, \\ \dot{x}_3(t) &= -\frac{I_2 - I_1}{I_3} x_1(t)x_2(t) - \frac{\lambda_3(t)}{I_3^2}, \\ \dot{\lambda}_1(t) &= \frac{I_1 - I_3}{I_2} x_3(t)\lambda_2(t) + \frac{I_2 - I_1}{I_3} x_2(t)\lambda_3(t), \\ \dot{\lambda}_2(t) &= \frac{I_3 - I_2}{I_1} x_3(t)\lambda_1(t) + \frac{I_2 - I_1}{I_3} x_1(t)\lambda_3(t), \\ \dot{\lambda}_3(t) &= \frac{I_3 - I_2}{I_1} x_2(t)\lambda_1(t) + \frac{I_1 - I_3}{I_2} x_1(t)\lambda_2(t), \end{aligned}$$

with the boundary conditions (26), and the optimal control laws are given by

$$u_i^*(t) = -\frac{\lambda_i(t)}{I_i}, \quad i = 1, 2, 3, \quad t \in [0, 100]. \tag{28}$$

By using the proposed algorithm with $\epsilon = 10^{-6}$, we reach the desired solution after $m = 6$ iterations.

Table 6: Simulation results for $n = 10$, Example 3.

m	J_m	\bar{E}_m	α_1	α_2	α_3
1	0.00468914	-	0.7314359	0.3851119	0.1356719
2	0.00468570	7.3285E-04	0.7453951	0.3581385	0.1272553
3	0.00468711	2.9978E-04	0.7437911	0.3617697	0.1281838
4	0.00468791	1.7023E-04	0.7437022	0.3619104	0.1291345
5	0.00468782	1.9383E-05	0.7437007	0.3619099	0.1290549
6	0.00468781	7.9781E-07	0.7437039	0.3619034	0.1290513

The simulation results along with the estimation of unknown parameters $\alpha_i, i = 1, 2, 3$, are given in Table 6.

Table 7: The maximum error of $x_1(t)$ in comparison to HPM [21], Example 3.

m		HPM		IVIM	
(Itr.)	Max Error		CPU time		CPU time
2	7.3102E-05		7.328		0.031
3	9.2612E-06		12.542		0.109
4	1.9301E-06		18.664		0.188
5	2.2560E-07		36.729		0.655
6	3.1420E-08		46.401		3.994

To compare the solutions with HPM [21], we have summarized the maximum error of $x_1(t)$ in both methods compared to “dsolve” package of Maple as the exact solution, In Table 7. As is evident, the IVIM is faster and

more accurate than HPM, so as to reach the error of 5×10^{-8} , HPM needs $m = 6$ iterations with 46.401 seconds of calculation time, but IVIM, needs 4 iterations with 0.188 seconds of calculation time.

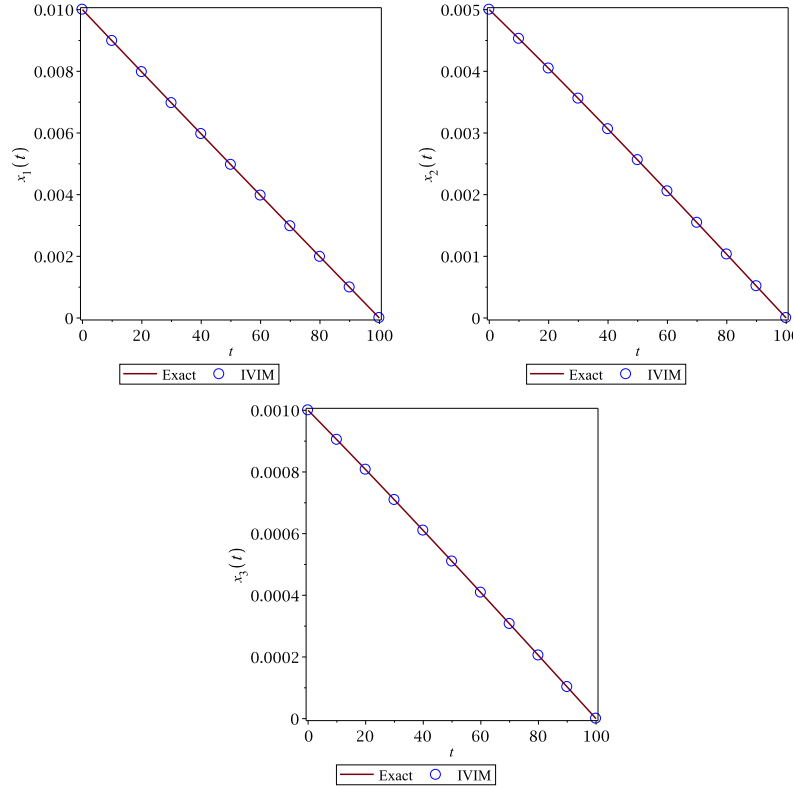


Figure 3: The optimal states $x_1(t), x_2(t)$, and $x_3(t)$, Example 3.

Figures 3–4 show the optimal states and controls of IVIM for $m = 6$ iterations compared to “dsolve” solutions. The maximum errors of our proposed method are 1.6156×10^{-8} , 1.0231×10^{-8} , and 1.39963×10^{-8} , for state functions x_1 , x_2 , and x_3 , respectively, and 3.4660×10^{-7} , 6.8377×10^{-7} , and 2.1526×10^{-7} , for control functions u_1 , u_2 , and u_3 , respectively. Meanwhile, only 3.994 seconds are needed to perform this operation.

5 Conclusions

In this paper, the interpolated VIM was used to solve NOCPs. This method used basic spline functions as interpolants, which is desirable for the approximate calculation of state and control functions. This method did not require

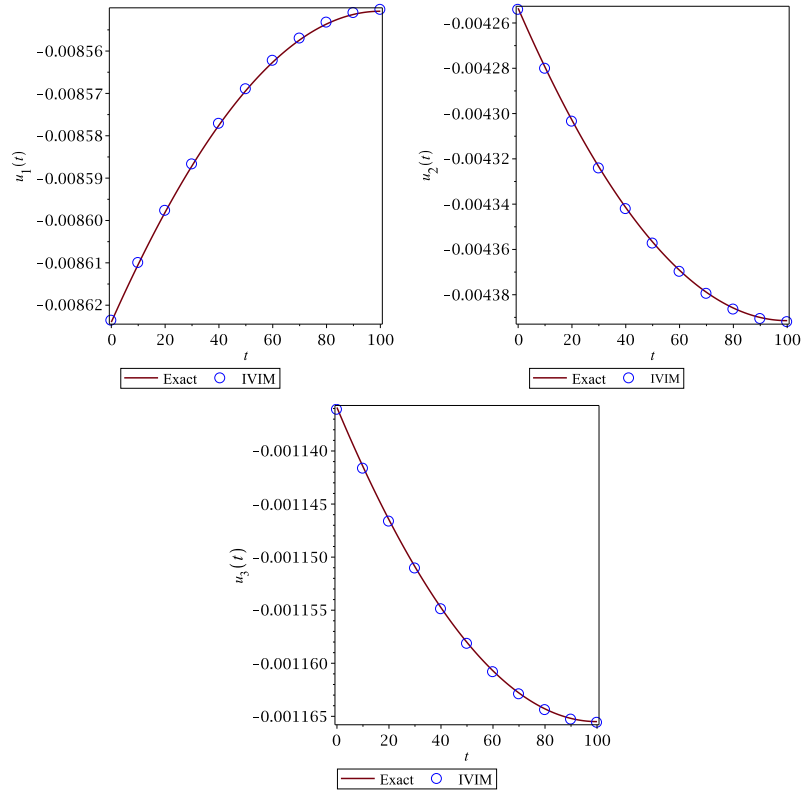


Figure 4: The optimal controls $u_1(t), u_2(t)$, and $u_3(t)$, Example 3.

complex analytical integrations and therefore was faster than VIM. Furthermore, the convergence range of the IVIM was wider, compared to the new SAM. In each iteration of the method, the co-state initial value was quickly calculated with an efficient technique described in the Examples section. Finally, the efficiency and reliability of the method were discussed, and the “dsolve” package of Maple 18 was used as the exact solution to check the validity of the method.

Acknowledgments

This research is supported by a grant from University of Neyshabur (Grant No. s/96/5173).

References

- [1] Ali, A., Iqbal, Q., Asamoah, J. K. K., and Islam, S. *Mathematical modeling for the transmission potential of Zika virus with optimal control strategies*. Eur. Phys. J. Plus 137 (1) (2022), 1–30.
- [2] Anjum, N., and He, J.-H. *Laplace transform: making the variational iteration method easier*. Appl. Math. Lett. 92 (2019), 134–138.
- [3] Ashchepkov, L., Dolgy, D., Kim, T., and Agarwal, R. *Optimal Control*. Springer International Publishing, 2022.
- [4] Berkani, S., Manseur, F., and Maida, A. *Optimal control based on the variational iteration method*. Comput. Math. Appl. 64 (4) (2012), 604–610.
- [5] El-Sayed, T. A., and El-Mongy, H. H. *A new numeric–symbolic procedure for variational iteration method with application to the free vibration of generalized multi-span Timoshenko beam*. J. Vib. Control 28 (7-8) (2022), 799–811.
- [6] Ermakov, A., Marie, A., and Ringwood, J. V. *Optimal control of pitch and rotational velocity for a cyclorotor wave energy device*. IEEE Trans. Sustain. Energy 13 (3) (2022), 1631–1640.
- [7] Gonzalez-Gaxiola, O., Biswas, A., Ekici, M., and Khan, S. *Highly dispersive optical solitons with quadratic–cubic law of refractive index by the variational iteration method*. J. Opt. 51 (1) (2022), 29–36.
- [8] He, J.-H. *Variational iteration method—a kind of non-linear analytical technique: some examples*. Int. J. Non-Linear Mech. 34 (4) (1999), 699–708.
- [9] Jafari, H., Ghasempour, S., and Baleanu, D. *On comparison between iterative methods for solving nonlinear optimal control problems*. J. Vib. Control 22 (9) (2016), 2281–2287.
- [10] Jajarmi, A., Pariz, N., Kamyad, A. V., and Effati, S. *A novel modal series representation approach to solve a class of nonlinear optimal control problems*. Int. J. Innov. Comput. Inf. Control. 1 (2011), 2.
- [11] Kim, Y., and Singh, T. *Energy-time optimal control of wheeled mobile robots*. J. Frankl. Inst. (2022).
- [12] Kincaid, D., Kincaid, D. R., and Cheney, E. W. *Numerical analysis: mathematics of scientific computing*, vol. 2. American Mathematical Soc., 2009.

- [13] Liu, S., You, H., Liu, Y., Feng, W., and Fu, S. *Research on optimal control strategy of wind-solar hybrid system based on power prediction*. ISA trans.123 (2022), 179–187.
- [14] Lohéac, J., Varma, V. S., and Morărescu, I. C. *Time optimal control for a mobile robot with a communication objective*. Math. Comput. Simul. 201 (2022), 96–120.
- [15] Ma, Z., and Zou, S. *Optimal Control Theory: The Variational Method*. Springer Singapore, 2021.
- [16] Majumder, M., Tiwari, P. K., and Pal, S. *Impact of saturated treatments on HIV-TB dual epidemic as a consequence of COVID-19: Optimal control with awareness and treatment*. Nonlinear Dyn. (2022), 1–34.
- [17] Matinfar, M., and Saeidy, M. *A new analytical method for solving a class of nonlinear optimal control problems*. Optim. Control Appl. Methods 35 (3) (2014), 286–302.
- [18] Meng, L., Kexin, M., Ruyi, X., Mei, S., and Cattani, C. *Haar wavelet transform and variational iteration method for fractional option pricing models*. Math. Methods Appl. Sci. (2022).
- [19] Mirhosseini-Alizamini, S. M., and Effati, S. *An iterative method for sub-optimal control of a class of nonlinear time-delayed systems*. Int. J. Control 92 (12) (2019), 2869–2885.
- [20] Nadeem, M., and He, J.-H. *He-Laplace variational iteration method for solving the nonlinear equations arising in chemical kinetics and population dynamics*. J. Math. Chem. 59 (5) (2021), 1234–1245.
- [21] Saberi Nik, H., Effati, S., Motsa, S. S., and Shirazian, M. *Spectral homotopy analysis method and its convergence for solving a class of nonlinear optimal control problems*. Numer. Algorithms 65 (1) (2014), 171–194.
- [22] Saberi Nik, H., Effati, S., and Yildirim, A. *Solution of linear optimal control systems by differential transform method*. Neural. Comput. Appl. 23 (5) (2013), 1311–1317.
- [23] Salkuyeh, D. K., and Tavakoli, A. *Interpolated variational iteration method for initial value problems*. Appl. Math. Model. 40 (5-6) (2016), 3979–3990.
- [24] Shirazian, M., and Effati, S. *Solving a class of nonlinear optimal control problems via he’s variational iteration method*. Int. J. Control Autom. Syst 10 (2) (2012), 249–256.
- [25] Shirazian, M., and Effati, S. *A novel successive approximation method for solving a class of optimal control problems*. Caspian Journal of Mathematical Sciences (CJMS) peer 9 (1) (2020), 124–136.

- [26] Singh, G., and Singh, I. *Semi-analytical solutions of three-dimensional (3d) coupled Burgers' equations by new Laplace variational iteration method*. Partial Differential Equations in Applied Mathematics (2022), 100438.
- [27] Sun, J.-S. *Approximate analytic solution of the fractal Fisher's equation via local fractional variational iteration method*. Therm. Sci. 26 (3 Part B) (2022), 2699–2705.
- [28] Tao, Z.-L., Chen, G.-H., and Chen, Y.-H. *Variational iteration method with matrix Lagrange multiplier for nonlinear oscillators*. J. Low Freq. Noise Vib. Act. Control. 38 (3-4) (2019), 984–991.

How to cite this article

Shirazian, M., The interpolated variational iteration method for solving a class of nonlinear optimal control problems. *Iran. j. numer. anal. optim.*, 2023; 13(2): 224-242. <https://doi.org/10.22067/ijnao.2022.77393.1158>